Machine Vision and Applications (2005)
Digital Object Identifier (DOI) 10.1007/s00138-004-0160-7

Machine Vision and
Applications

Computing depth maps from descent images

Yalin Xiong[1], **Clark F. Olson**[2], **Larry H. Matthies**[3]

[1] KLA-Tencor, 160 Rio Robles St., San Jose, CA 95134, USA
[2] University of Washington, Computing and Software Systems, 18115 Campus Way NE, Box 358534, Bothell, WA 98011, USA
[3] Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Drive, M/S 125-209, Pasadena, CA 91109, USA

Received: 7 November 2002 / Accepted: 13 September 2004
Published online: 25 February 2005 – © Springer-Verlag 2005

Abstract. In the exploration of the planets of our solar system, images taken during a lander's descent to the surface of a planet provide a critical link between orbital images and surface images. The descent images not only allow us to locate the landing site in a global coordinate frame, but they also provide progressively higher-resolution maps for mission planning. This paper addresses the generation of depth maps from the descent images. Our approach has two steps, motion refinement and depth recovery. During motion refinement, we use an initial motion estimate in order to avoid the intrinsic motion ambiguity. The objective of the motion-refinement step is to adjust the motion parameters such that the reprojection error is minimized. The depth-recovery step correlates adjacent frames to match pixels for triangulation. Due to the descending motion, the conventional rectification process is replaced by a set of anti-aliasing image warpings corresponding to a set of virtual parallel planes. We demonstrate experimental results on synthetic and real descent images.

Keywords: Planetary exploration – Structure from motion – Descent images – Motion estimation – Terrain mapping

1 Introduction

Future space missions that land on Mars (and other planetary bodies) may include a downward-looking camera mounted on the vehicle as it descends to the surface. Images taken by such a camera during the descent provide a critical link between orbital images and lander/rover images on the surface of the planet. By matching the descent images against orbital images, the descent vehicle can localize itself in global coordinates and, therefore, achieve precision landing. Through analysis of the descent images, we can build a multiresolution terrain map for safe landing, rover planning, navigation, and localization. This paper addresses the issue of generating multiresolution terrain maps from a sequence of descent images. We use motion-estimation and structure-from-motion techniques to recover depth maps from the images. A new technique for computing depths is described that is based on correlating the images after performing anti-aliasing image warpings that correspond to a set of virtual planar surfaces.

It is well known that, when the viewed terrain is a planar surface, the motion-recovery problem is ill-posed, since translations parallel to the surface appear similar to rotations about axes parallel to the surface. Motion recovery is, therefore, not generally reliable for this scenario. However, if we have an independent means to measure the orientation of the camera, we can obtain stable motion recovery. For planetary exploration missions, such measurements can be provided by the inertial navigation sensors on the landing spacecraft.

For space missions, it is likely that the motion will be nearly perpendicular to the planetary surface. For the Mars Polar Lander mission, which was unable to return data due to loss of the lander, it was planned that the camera would take an image every time the distance to the ground halved. In other words, there would be roughly a scale factor of two between adjacent frames in the sequence. A similar scenario is likely in future missions. The large change of scale prohibits us from tracking many features across several frames. For most features, we limit our correlation and depth recovery to adjacent frames for this reason.

The descending motion also causes problems in correlating the images. Since the epipoles are located near the center of the images, it is not practical to rectify adjacent frames in the same manner that traditional stereo techniques do. Instead, we perform a virtual rectification, resampling the images by considering a set of parallel planar surfaces through the terrain. Each surface corresponds to a projective warping between the adjacent images. The surface that yields the best correlation at each pixel determines the depth estimate for that location. This methodology not only aligns images according to the epipolar lines but also equalizes the image scales using anti-aliased warpings.

Of course, many other approaches have been proposed for recovering motion and depth from image sequences [2,4,6,8,11,14]. This work differs from most previous work in two ways. First, almost all of the motion is forward along

Correspondence to: Clark F. Olson
(e-mail: cfolson@u.washington.edu,
Tel.: +1-425-3525288, Fax: +1-425-3525216)

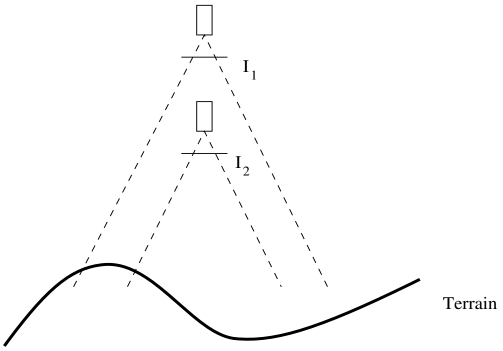

Fig. 1. Descent motion

the camera pointing direction. Second, large movements in the camera position occur between frames, usually doubling the resolution of the images at each frame. This work is thus in part, an application of previous work to the problem of mapping spacecraft descent imagery and, in part, new techniques for dealing with the above problems. The technique produces dense maps of the terrain and operates under the full perspective projection.

In the next two sections, we describe the motion-refinement and depth-recovery steps in detail. We then discuss our experiments on synthetic and real descent images. The results demonstrate the various terrain features that can be recovered. Near the landing site, small obstacles such as rocks and gullies can be identified for planning local rover navigation. Further from the landing site, larger features such as mountain slopes and cliffs are visible for use in long-range planning.

2 Motion refinement

Recovering camera motion from two or more frames is one of the classical problems in computer vision. Linear [7] and nonlinear [13] solutions have been proposed. Our scenario involves a downward motion towards a roughly planar surface (as in Fig. 1). Generic motion recovery from matched features is ill-posed owing to a numerical singularity (rotations and translations are difficult to distinguish.) Since the camera can be rigidly attached to the lander, and the change in the lander orientation can be measured accurately by an inertial navigation system onboard, we can eliminate the singularity problem by adding a penalty term for deviating from the measured orientation. The following two subsections briefly explain our feature correspondence and nonlinear optimization for motion refinement.

2.1 Feature correspondence

For each pair of adjacent frames in the sequence, we determine correspondences for features that have been selected in the higher-resolution frame into the lower-resolution frame. Forstner's interest operator [3] is used to evaluate the distinctiveness of the features in the higher-resolution frame. We select the features with high scores while disallowing features that are too close together (Fig. 2a).

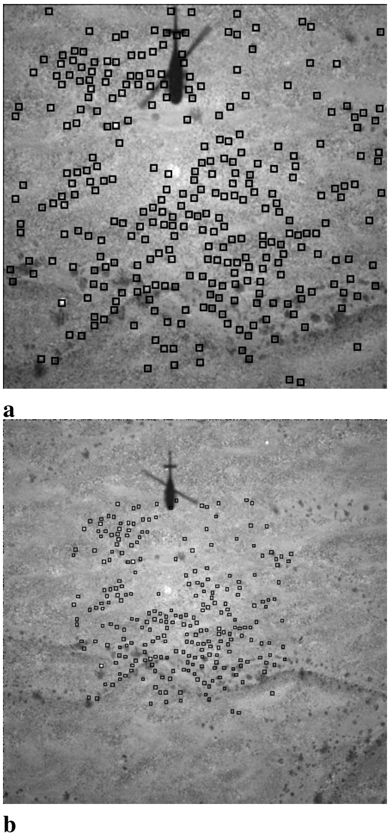

Fig. 2a,b. Feature correspondence between adjacent frames. **a** Selected features. **b** Correspondences detected

Once the image resolutions have been equalized (using downsampling or anti-aliasing warping, if necessary), feature correspondence can be determined in a straightforward manner. For every feature in the reference image, we search an area in the target image for a match. The location of the search area is derived from the initial estimate of the vehicle ego-motion and its altitude. The initial estimates do not need to be precise. The size of the search area is determined by how uncertain the initial estimates are. Once the search area is located, we detect the feature match through normalized correlation.

2.2 Nonlinear motion estimation

The objective of motion refinement is to establish the precise camera motion between two adjacent frames such that the epipolar constraints are satisfied to subpixel accuracy. It is unrealistic to expect the onboard inertial sensors to track the camera orientation with such precision. It is therefore crucial to be able to refine the motion parameters prior to recovering the depth map.

The matched features provide a rich set of observations to constrain the camera motion, even though the relationship between the locations of the matched features and the camera motion parameters is nonlinear. Let us assume that the projection matrix of the camera (including the calibrated internal parameters) is $\vec{A}$, the location of feature i at time t is $[X_i^t, Y_i^t, Z_i^t]^T$ in the camera frame of reference, its image location at time t represented in homogeneous coordinates is

$[x_i^t, y_i^t, z_i^t]^T$, and the camera motion between time t and time $t+1$ is composed of a translation $\vec{T}$ and rotation $\vec{R}$ (3×3 matrix). The projection of the feature at time t is:

$$\begin{bmatrix} x_i^t \\ y_i^t \\ z_i^t \end{bmatrix} = \vec{A} \begin{bmatrix} X_i^t \\ Y_i^t \\ Z_i^t \end{bmatrix}, \tag{1}$$

and the projection at time $(t+1)$ is:

$$\begin{bmatrix} x_i^{t+1} \\ y_i^{t+1} \\ z_i^{t+1} \end{bmatrix} = \vec{A} \begin{bmatrix} X_i^{t+1} \\ Y_i^{t+1} \\ Z_i^{t+1} \end{bmatrix} = \vec{A} \left(\vec{R} \begin{bmatrix} X_i^t \\ Y_i^t \\ Z_i^t \end{bmatrix} + \vec{T} \right). \tag{2}$$

Therefore, the feature motion in the image is:

$$\begin{aligned} \begin{bmatrix} x_i^{t+1} \\ y_i^{t+1} \\ z_i^{t+1} \end{bmatrix} &= \vec{A} \left(\vec{R}\vec{A}^{-1} \begin{bmatrix} x_i^t \\ y_i^t \\ z_i^t \end{bmatrix} + \vec{T} \right) \\ &= \vec{U} \begin{bmatrix} x_i^t \\ y_i^t \\ z_i^t \end{bmatrix} + \vec{V}, \end{aligned} \tag{3}$$

where $\vec{U} = \vec{A}\vec{R}\vec{A}^{-1}$ is a 3×3 matrix and $\vec{V} = \vec{A}\vec{T}$ is a 3-vector. Let $[c_i^t, r_i^t] = [x_i^t/z_i^t, y_i^t/z_i^t]$ denote the actual column and row location of feature i in image coordinates at time t. We then have the predicted feature locations at time $t+1$ as:

$$\hat{c}_i^{t+1} = \frac{u_{00}x_i^t + u_{01}y_i^t + u_{02}z_i^t + v_0}{u_{20}x_i^t + u_{21}y_i^t + u_{22}z_i^t + v_2}, \tag{4}$$

$$\hat{r}_i^{t+1} = \frac{u_{10}x_i^t + u_{11}y_i^t + u_{12}z_i^t + v_1}{u_{20}x_i^t + u_{21}y_i^t + u_{22}z_i^t + v_2}, \tag{5}$$

where u_{ij} and v_i are elements of $\vec{U}$ and $\vec{V}$, respectively.

There are two ways to optimize the camera motions in the above equations. One is to reduce the two equations into one by eliminating z_i^t. We would then minimize the summed deviation from the equation specifying a nonlinear relation between $[c_i^t, r_i^t]$ and $[\hat{c}_i^{t+1}, \hat{r}_i^{t+1}]$. Though this method is concise and simple, it poses a problem in the context of least-squares minimization in that the objective function does not have a physical meaning.

The other approach to refine the motion estimate is to augment the parameters with depth estimates for each of the features. There are two advantages to this approach. First, the objective function becomes the distance between the predicted and observed feature locations, which is a meaningful measure for optimization. In addition, in the context of mapping descent images, we have a good initial estimate of the depth value from the spacecraft altimeter. Incorporating this information will thus improve the optimization in general.

Let us say that the depth value of feature i at time t is d_i^t and the camera is pointing along the z-axis; the homogeneous coordinates of the feature are $[x_i^t, y_i^t, z_i^t]^T = d_i^t[c_i^t, r_i^t, 1]^t$. Therefore, the overall objective function we are minimizing is:

$$\sum_{i=1}^{N} \left(\left(c_i^{t+1} - \hat{c}_i^{t+1}\right)^2 + \left(r_i^{t+1} - \hat{r}_i^{t+1}\right)^2 \right), \tag{6}$$

where N is the number of features and $\hat{c}_i^{t+1}$ and $\hat{r}_i^{t+1}$ are nonlinear functions of the camera motion and depth value d_i^t given by Eqs. 4 and 5. We perform nonlinear minimization using the Levenberg–Marquardt algorithm with the estimated position as the starting point. Robustness is improved through the use of robust statistics in the optimization.

The result of the optimization is a refined estimate of the rotation R and translation T between the camera positions. In the optimization, we represent the rotation using a quaternion. In order to resolve between translations and rotations, which appear similar, a penalty term is added to the objective function Eq. 6 that prefers motions close to the initial motion estimate. This constrains the motion not to deviate far from the orientation estimated by navigation sensors while allowing for some deviation in order to fit the observed data. An additional penalty term is added that forces the quaternion to have unit length and, thus, correspond to a valid rotation matrix.

Equation 6 specifies the objective function for two adjacent images. A long sequence of descending images requires a common scale reference in order to build consistent multiresolution depth maps. The key to achieving this is to track features over more than two images. From Eq. 3 the depth value of feature i at time $t+1$ can be represented as

$$d_i^{t+1} \begin{bmatrix} c_i^{t+1} \\ r_i^{t+1} \\ 1 \end{bmatrix} = \vec{U} d_i^t \begin{bmatrix} c_i^t \\ r_i^t \\ 1 \end{bmatrix} + \vec{V}. \tag{7}$$

Thus, the overall objective is to minimize the sum of Eq. 6 for all adjacent pairs while maintaining the consistent scale reference by imposing the constraint in Eq. 7 for all features tracked over more than two frames.

3 Depth map recovery

The second step of our method generates depth maps by performing correlations between image pairs. In order to compute the image correlation efficiently, we could rectify the images in a manner similar to binocular stereo. Unfortunately, we cannot simply rectify the images along scanlines because the epipolar lines intersect near the center of the images. If we resample the images along epipolar lines, we will oversample near the image center and undersample near the image boundaries.

Alternative rectification methods have recently been proposed that improve handling of these issues [9,10]. However, these methods enlarge the images and resample unevenly. Another alternative is to not resample the images at all. Matches can be found along the epipolar lines even if they are not horizontal. This method, however, requires considerably more computation time.

In order to avoid these problems and perform the correlation efficiently, we adopt a slicing algorithm. The main idea is to use a set of virtual planar surfaces slicing through the terrain as shown in Fig. 3. A similar concept was used by Collins [2] to perform matching between features extracted from the images. See also [12].

The virtual planar surfaces are similar in concept to horopter surfaces [1] in stereo. For every planar surface k, if a terrain surface patch lies on the planar surface, then there exists a projective warping $\vec{P}_k$ between the two images for this patch. If we designate the first image $I_1(x,y)$ and the second image $I_2(x,y)$, then for every virtual planar surface, we can

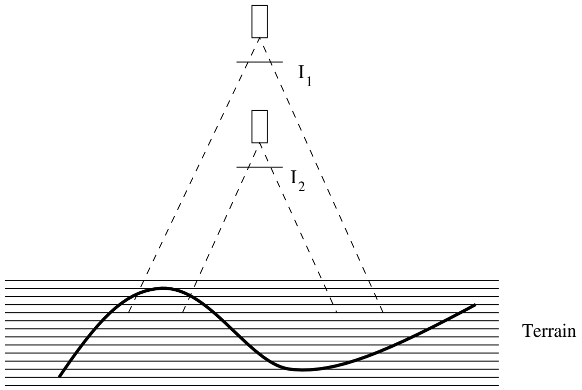

Fig. 3. The terrain is sliced with virtual parallel planes

compute the sum of squared differences (SSD) as:

$$C_k(x,y) = \sum_{m=x-W}^{x+W} \sum_{n=y-W}^{y+W} \left(I_1(m,n) - I_2^k(m,n)\right)^2, \quad (8)$$

where $2W + 1$ is the size of the correlation window and $I_2^k(x,y)$ is a warped version of $I_2(x,y)$:

$$I_2^k(x,y) = I_2\left(\frac{p_{00}x + p_{01}y + p_{02}}{p_{20}x + p_{21}y + p_{22}}, \frac{p_{10}x + p_{11}y + p_{12}}{p_{20}x + p_{21}y + p_{22}}\right), \quad (9)$$

where p_{ij} are elements of the 3×3 matrix $\vec{P}_k$. Due to the large resolution difference, an anti-aliasing resampling [5] or a uniform downsampling of $I_2(x,y)$ is applied before the image warping. In practice, if the camera heading directions are almost perpendicular to the ground, a uniform downsampling before warping is sufficient. Otherwise, a space-variant downsampling should be used to equalize the image resolutions.

The estimated depth value at each pixel is the depth of the plane z_k whose corresponding SSD image pixel $C_k(x,y)$ is the smallest:

$$z(x,y) = z_k, \quad (10)$$

where

$$C_k(x,y) \leq C_j(x,y), j = 1, \ldots, M, \quad (11)$$

and M is the number of planar surfaces. To further refine the depth values, the underlying SSD curve can be interpolated by a quadratic curve and the "subpixel" depth value computed [15] as:

$$z(x,y) = z_k + \frac{\delta z(C_{k+1}(x,y) - C_{k-1}(x,y))}{2(C_{k+1}(x,y) + C_{k-1}(x,y) - 2C_k(x,y))}, \quad (12)$$

where δz is the depth increment between adjacent planar surfaces. In order to improve the localization of this operation, we compute the SSD between the windows with a Gaussian modulation function so that the pixels closer to the center of the window have more weight than the pixels at the edge of the window.

The projective warping matrix $\vec{P}_k$ is derived from the parameters of the camera motion and the planar surfaces. For an arbitrary point $\vec{X}$ in some reference frame, its projection is expressed as $\vec{x} = \vec{M}(\vec{X} - \vec{C})$, where $\vec{C}$ is the position of the camera nodal point and $\vec{M}$ is the projection matrix. Note that $\vec{C}$ and $\vec{M}$ encapsulate the camera motion between the images, since they are represented in a common reference frame. Let $\vec{C}_1$ and $\vec{M}_1$ represent the higher camera, $\vec{C}_2$ and $\vec{M}_2$ represent the lower camera in Fig. 3.[1] $\vec{N}^T\vec{X} + z_k = 0$ represents the set of planar surfaces. For any pixel in image 2 (i.e., the lower camera), its location must lie on a 3D ray:

$$\vec{X} = s\vec{M}_2^{-1}\begin{bmatrix} c_2 \\ r_2 \\ 1 \end{bmatrix} + \vec{C}_2, \quad (13)$$

where c_2 and r_2 are, respectively, the column and row location of the pixel and s is a positive scale factor. If the pixel is from a point on the planar surface, then the following constraint must be satisfied:

$$s\vec{N}^T\vec{M}_2^{-1}\begin{bmatrix} c_2 \\ r_2 \\ 1 \end{bmatrix} + \vec{N}^T\vec{C}_2 + z_k = 0. \quad (14)$$

Therefore, the scale factor s must be

$$s = -\frac{\vec{N}^T\vec{C}_2 + z_k}{\vec{N}^T\vec{M}_2^{-1}[c_2, r_2, 1]^T}. \quad (15)$$

We can then reproject the point onto the first image using Eqs. 13 and 15:

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = \vec{M}_1(\vec{X} - \vec{C}_1) = \vec{P}_k\begin{bmatrix} c_2 \\ r_2 \\ 1 \end{bmatrix}, \quad (16)$$

where $\vec{P}_k$ is a 3×3 matrix specifying the projective warping:

$$\vec{P}_k = \vec{M}_1(\vec{C}_2 - \vec{C}_1)\vec{N}^T\vec{M}_2^{-1} - (\vec{N}^T\vec{C}_2 + z_k)\vec{M}_1\vec{M}_2^{-1}. \quad (17)$$

Note that the depth recovery is numerically unstable in the vicinity of the epipoles, located near the center of the image. Pixels near the epipoles usually have a small amount of parallax, even with large camera motions. The SSD curves in these areas are very flat, and, thus, accurate depth recovery is difficult. These regions can be easily filtered, if desired, by imposing a minimum curvature threshold at the minima of the SSD curves.

4 Experiments

Figure 4a–c shows a synthetic set of nested descent images (400×400 pixels). For this set of images, the terrain model is composed of a slowly varying terrain surface overlaid with rocks distributed according to a statistical model. The height of the camera decreases from 25 m above the ground to 6 m above the ground. The field of view of the camera is 70°.

Figures 4d,e show a visualization of the depth maps, with the image draped over the terrain. The maps have root-mean-square errors of 4.6 cm and 9.7 cm, respectively. Note that the

[1] In the notation of the previous section, $\vec{M}_1 = \vec{A}$, $\vec{C}_1 = 0$, $\vec{M}_2 = \vec{A}R$, and $\vec{C}_1 = -R^{-1}T$. The change of notation is primarily for convenience.

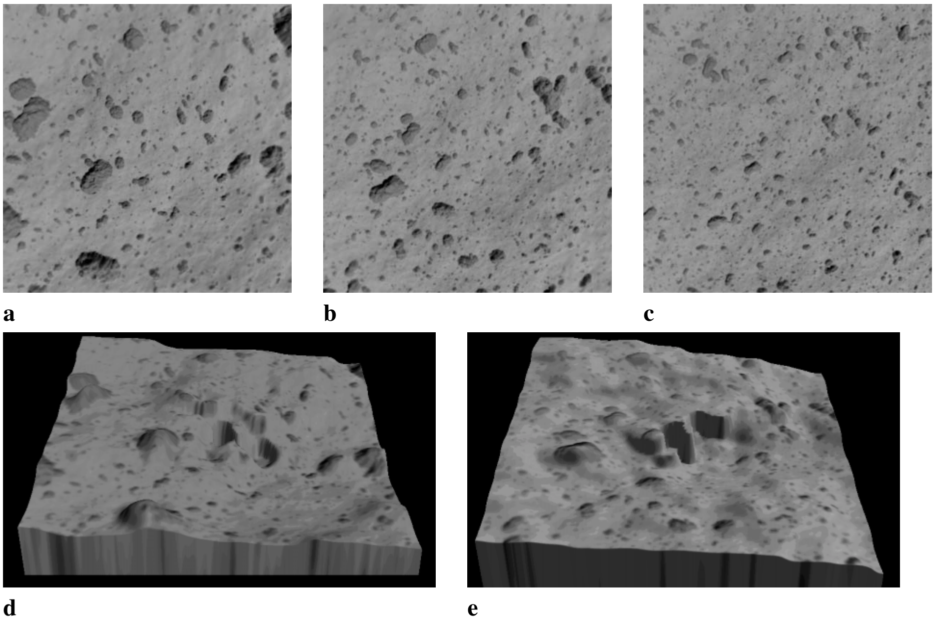

Fig. 4a–e. Synthetic descent image sequence. **a** Image at 6 m elevation. **b** Image at 12 m elevation. **c** Image at 25 m elevation. **d** Rendered terrain map at 6 m elevation. **e** Rendered terrain map at 12 m elevation

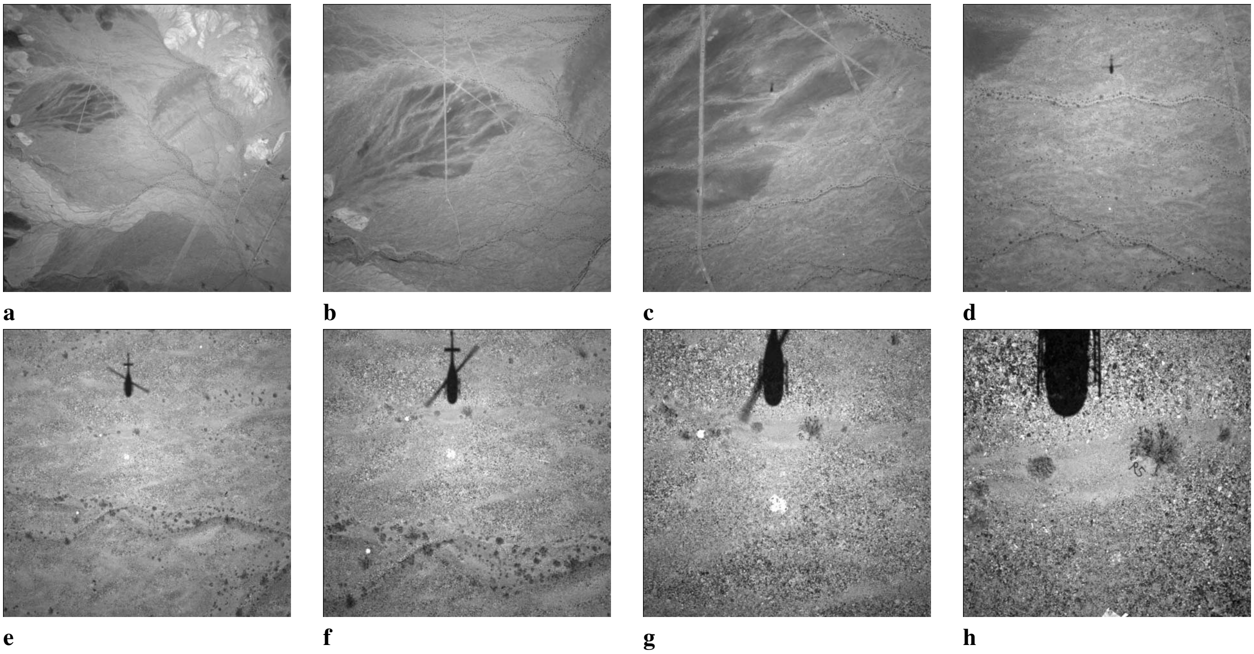

Fig. 5a–h. Descent sequence captured with a helicopter. **a** Elevation: 1085 m. **b** Elevation: 590 m. **c** Elevation: 238 m. **d** Elevation: 128 m. **e** Elevation: 55 m. **f** Elevation: 32 m. **g** Elevation: 15 m. **h** Elevation: 8 m

areas close to the focus of expansion (at the center of the image) have larger error than the rest of the image, owing to the geometrical instability at the focus of expansion. In both image pairs, the general downward slope of the terrain from back to front and left to right can be observed. In addition, individual rocks can be distinguished, particularly in the lower-elevation image pair. These techniques have been tested extensively on similar synthetic images with similar results.

For these experiments, we generated our initial estimates of the camera motion by perturbing the actual camera values by a random noise of magnitude 2°. This level of accuracy in the orientation can be achieved by the onboard inertial navigation system during an actual landing. The overall quality of the recovered depth maps appears adequate for navigation in the vicinity of the landing and long-range planning of goals far from the landing site.

Our techniques were also tested using a set of descent images collected in the desert area near Silver Lake, CA using a helicopter. Figure 5 shows eight frames from this sequence. The initial camera motions were estimated using control points

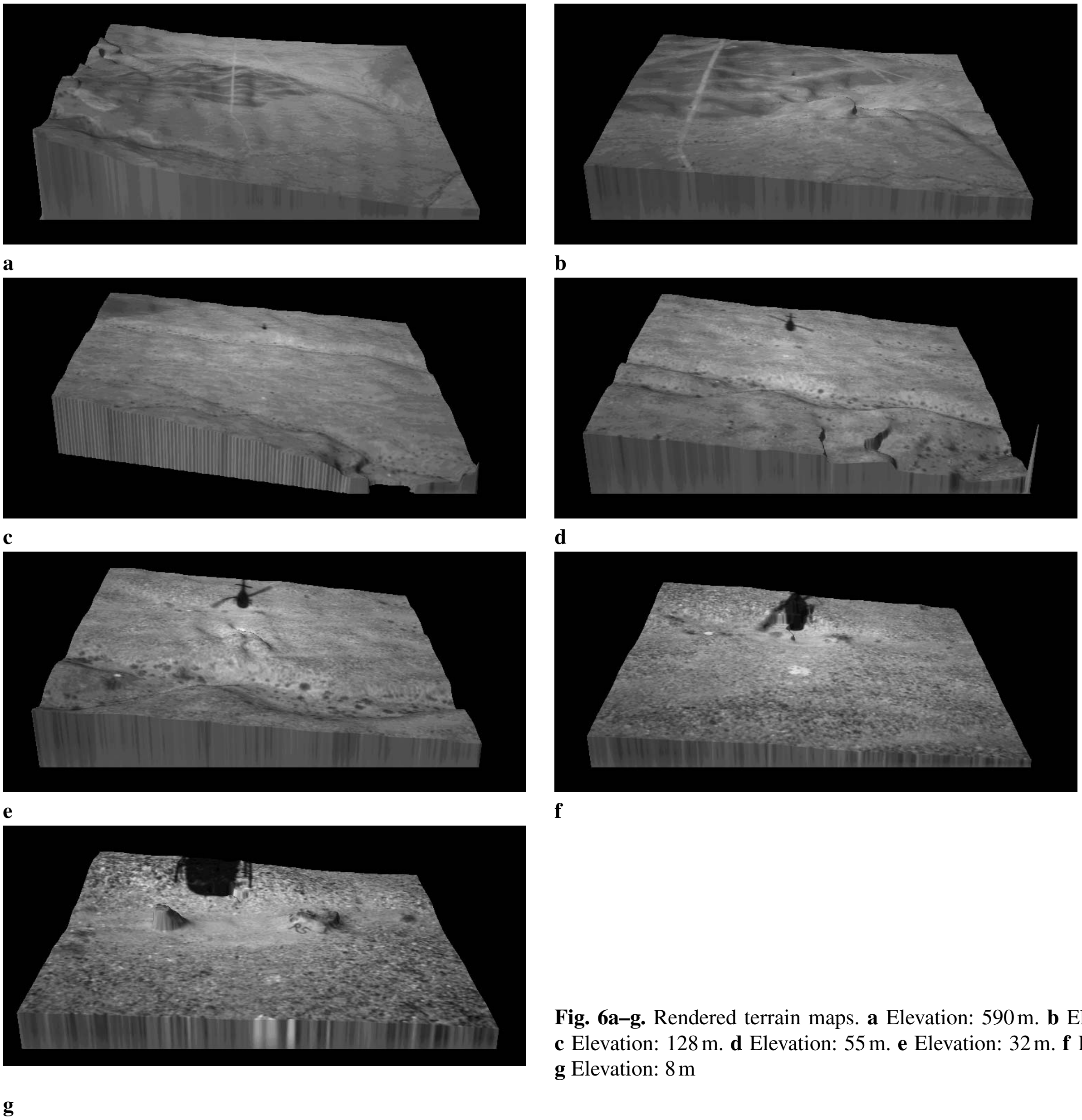

Fig. 6a–g. Rendered terrain maps. **a** Elevation: 590 m. **b** Elevation: 238 m. **c** Elevation: 128 m. **d** Elevation: 55 m. **e** Elevation: 32 m. **f** Elevation: 15 m. **g** Elevation: 8 m

on the ground. Several of the images contain significant lateral motions due to the difficulty in maintaining the x-y position of the helicopter during the data collection. Figure 6 shows the images draped over the visualized terrain, and Fig. 7 shows the same data without texture mapping.

Since these are real images captured using a moving helicopter, the focus of expansion for each image pair is not at the center of the image (although it is reasonably close in Figs. 6b and e). In Figs. 6f and g, the focus of expansion can be seen above the center of the image, while it is near the bottom-right corner in Figs. 6c and d. In Fig. 6a, the focus of expansion is off of the image to the left. The instability can be seen in these locations where the rendered map becomes wavy or choppy. As the distance from the focus of expansion increases, the terrain elevations become more accurate. In Fig. 6c, the lower altitude image did not entirely overlap the higher altitude image, resulting in the lack of elevation data in the lower-left corner of the result for that image pair.

For the images in this data set, the terrain slopes downward from left to right, which can be observed in the rendered maps. Some of the interesting terrain features include the channels in Figs. 6c–e and the bushes visible in Fig. 6g. Note that the areas in which the helicopter shadow is present yield good results, despite the movement of the shadow. This can be attributed to the robust methods that we use for both motion estimation and template matching. Overall, this data set indicates that we can robustly compute maps that are useful for navigation over both small and large scales using real images, albeit under somewhat different conditions than would be encountered by an actual Mars lander.

While no descent images are currently available from other planets, the three final Ranger missions to the moon (Ranger 7 in 1964, Ranger 8 in 1965, and Ranger 9 in 1965) transmitted images on their descent to the surface. In fact, the mission of these spacecraft was to return images while crashing into the moon at flight velocity. Unfortunately, these images are

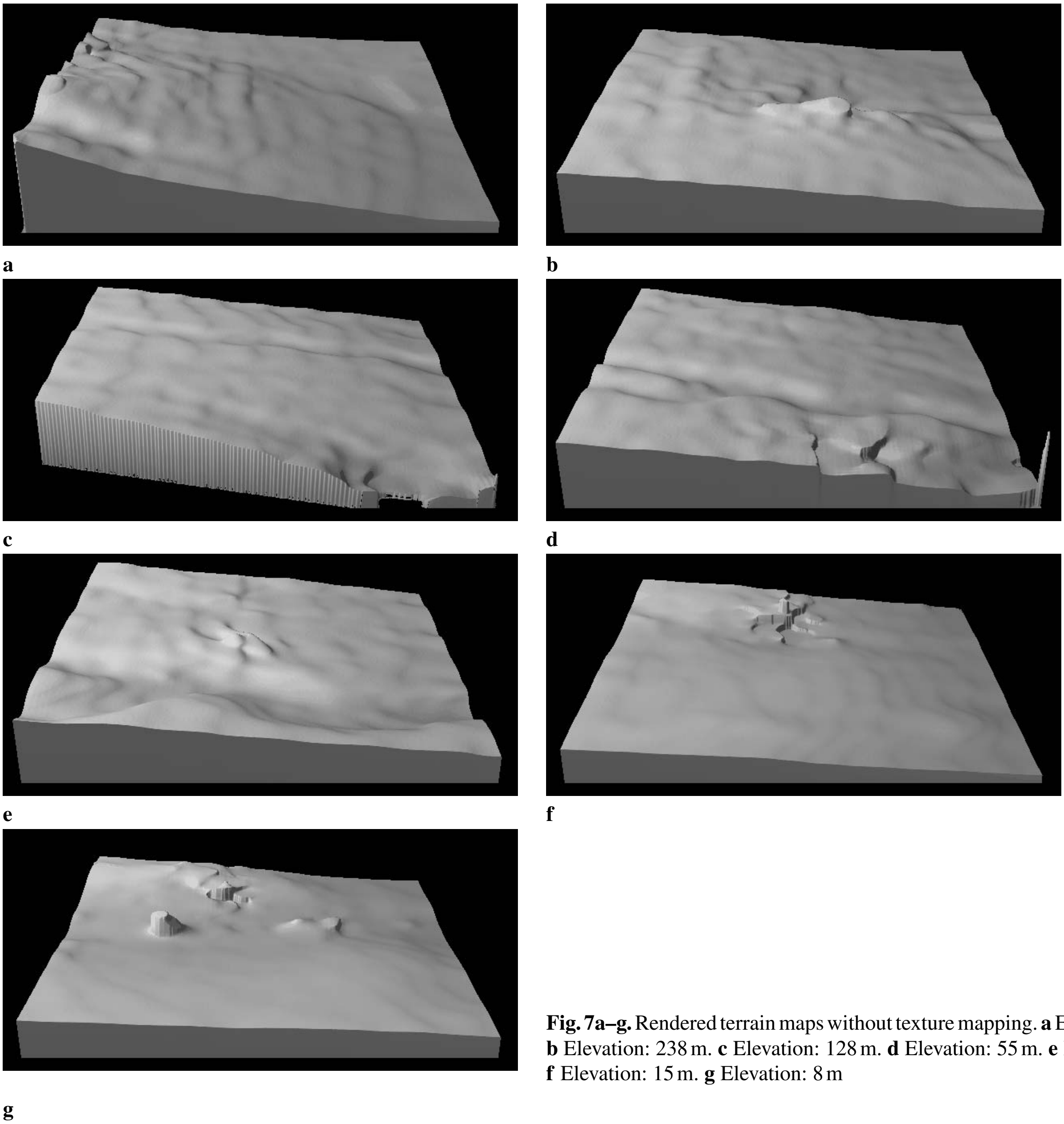

Fig. 7a–g. Rendered terrain maps without texture mapping. **a** Elevation: 590 m. **b** Elevation: 238 m. **c** Elevation: 128 m. **d** Elevation: 55 m. **e** Elevation: 32 m. **f** Elevation: 15 m. **g** Elevation: 8 m

difficult to work with, since the camera was uncalibrated and the images were transmitted in analog form. Figure 8 shows images from Ranger 7 and Ranger 9 and rendered terrain maps that show qualitative results with respect to the shape of the terrain. High-quality quantitative results are not possible with these data. Nevertheless, we are able to extract interesting data with respect to the slope of the terrain and certain depressions caused by impact craters. For the Ranger 7 image, the focus of expansion occurs at the upper-left corner of the image. For the Ranger 9 image, the focus of expansion occurs just above the large crater that is visible.

5 Summary

We have presented techniques for extracting depth maps from a sequence of descent images, such as those that would be acquired by a lander descending to a planetary surface. The method consists of two primary steps: motion estimation and depth recovery. Motion estimation is performed by matching features and minimizing a least-squares objective function using nonlinear methods. The depth map is then recovered using a novel technique where the terrain is sliced by virtual planes, similar to horopter surfaces in stereo. Each plane can be thought of as a vertical displacement. The plane yielding the lowest SSD is selected as the depth for each pixel, and subpixel estimation techniques are used to improve the estimate. We have performed experiments with this method on synthetic and real image sequences. The experiments have resulted in maps that can be used for navigation and planning at a scale roughly proportional to the distance from the landing site.

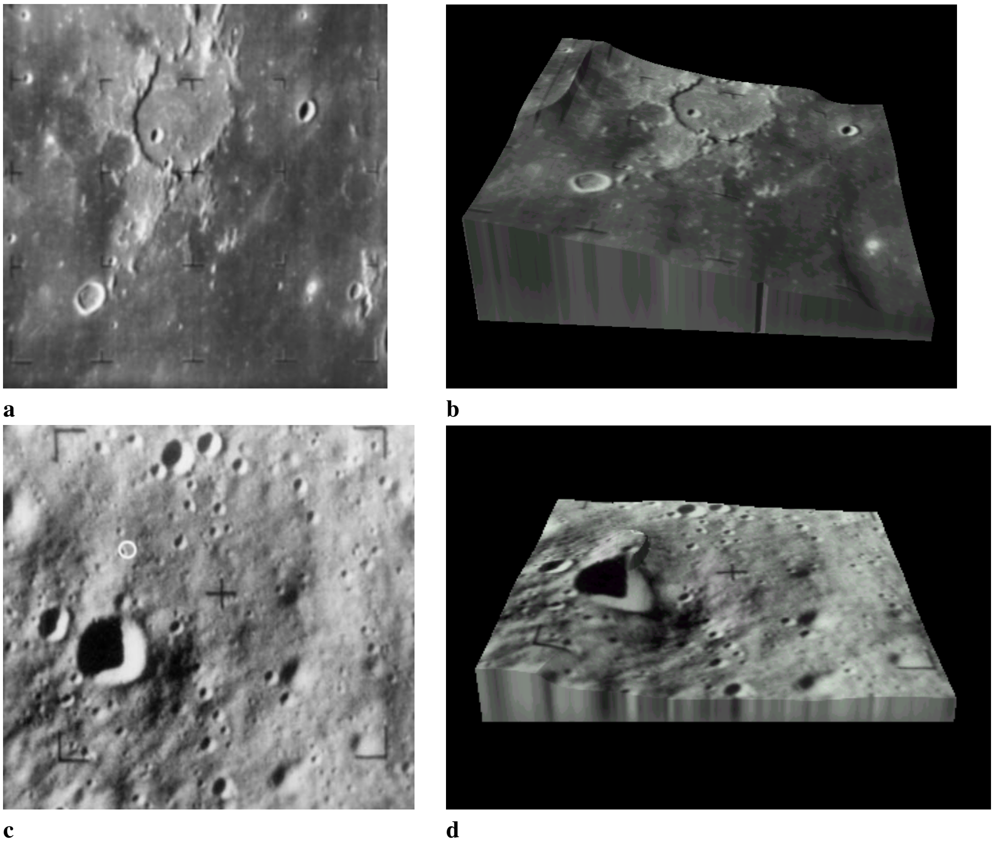

Fig. 8a–d. Moon images from Ranger missions. **a** Moon image from Ranger 7. **b** Rendered terrain map. **c** Moon image from Ranger 9. **d** Rendered terrain map.

Acknowledgements. The research described in this paper was carried out in part at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. An earlier version of this work appeared in the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition [16].

References

1. Burt PJ, Wixson L, Salgian G (1995) Electronically directed “focal’’ stereo. In: Proceedings of the international conference on computer vision, pp 94–101
2. Collins RT (1996) A space-sweep approach to true multi-image matching. In: Proceedings of the IEEE conference on computer Vision and pattern recognition, pp 358–363
3. Förstner W (1994) A framework for low-level feature extraction. In: Proceedings of the European conference on computer vision, pp 383–394
4. Hanna KJ (1991) Direct multi-resolution estimation of ego-motion and structure for motion. In: Proceedings of the IEEE workshop on visual motion, pp 156–162
5. Heckbert P (1986) Survey of texture mapping. IEEE Computr Graph Appl 6(11):56–67
6. Heeger DJ, Jepson AD (1992) Subspace methods for recovering rigid motion: I. Algorithm and implementation. Int J Comput Vis 7(2):95–117
7. Longuet-Higgins HC (1981) A computer algorithm for reconstructing a scene from two projections. Nature 293:133–135
8. Oliensis J, Genc Y (1999) Fast algorithms for projective multi-frame structure from motion. In: Proceedings of the international conference on computer vision, 1:536–543
9. Pollefeys M, Koch R, Van Gool L (1999) A simple and efficient rectification method for general motion. In: Proceedings of the international conference on computer vision, 1:496–501
10. Roy S, Meunier J, Cox IJ (1997) Cylindrical rectification to minimize epipolar distortion. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 393–399
11. Soatta S, Perona P (1998) Reducing “structure from motion”: a general framework for dynamic vision, part 1: Modeling. IEEE Trans Pattern Anal Mach Intell 20(9):933–942
12. Szeliski R, Golland P (1999) Stereo matching with transparency and matting. Int J Comput Vis 32(1):45–61
13. Szeliski R, Kang SB (1994) Recovering 3d shape and motion from image streams using non-linear least squares. J Vis Commun Image Represent 5(1):10–28
14. Tomasi C, Kanade T (1992) Shape and motion from image streams under orthography: a factorization method. Int J Comput Vis 9(2):137–154
15. Xiong Y. Matthies LH (1997) Error analysis for a real-time stereo system. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1087–1093
16. Xiong Y, Olson CF, Matthies LH (2001) Computing depth maps from descent imagery. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 1:392–397

Yalin Xiong is Director of Engineering in RAPID division of KLA-Tencor, specializing in image processing and computer vision algorithms for semiconductor inspection equipments. Prior to joining KLA-Tencor, he was a senior staff in Jet Propulsion Lab, Caltech from 1998 to 2000, and senior engineer in Apple Computer from 1996 to 1998. Dr. Yalin Xiong got Ph.D. from the Robotics Institute of Carnegie Mellon University, and B.S. from the University of Science and Technology of China.

Clark F. Olson received the B. S. degree in computer engineering in 1989 and the M.S. degree in electrical engineering in 1990, both from the University of Washington, Seattle. He received the Ph. D. degree in computer science in 1994 from the University of California, Berkeley. After spending two years doing research at Cornell University, he moved to the Jet Propulsion Laboratory, where he spent five years working on computer vision techniques for Mars rovers and other applications. Dr. Olson joined the faculty at the University of Washington, Bothell in 2001. His research interestes include computer vision and mobile robotics. He teaches classes on mathematical principles of computing, database systems, and computer vision. He continues to work with NASA/JPL on computer vision techniques for Mars exploration.

Larry H. Matthies PhD, computer science, Carnegie Mellon University, 1989; Supervisor, Machine Vision Group, Jet Propulsion Laboratory (JPL). Dr. Matthies has 21 years experience in developing perception systems for autonomous navigation of robotic ground and air vehicles. He pioneered the development of real-time stereo vision algorithms for range imaging and accurate visual odometry in the 1980's and early 1990's. These algorithms will be used for obstacle detection onboard the Mars Exploration Rovers (MER). The GESTALT system for obstacle avoidance for the MER rovers was developed in his group; this system is also currently the baseline for onboard obstacle detection for the 2009 Mars Science Laboratory (MSL) mission. His stereo vision-based range imaging algorithms were used for ground operations by the Mars Pathfinder mission in 1997 to map terrain with stereo imagery from the lander for planning daily operations for the Sojourner rover. He initiated the development at JPL of computer vision te s for autonomous safe landing and landing hazard avoidance for missions to Mars, asteroids, and comets. His group developed the Descent Image Motion Estimation System (DIMES) that will be used to estimate horizontal velocity of the MER landers during terminal descent. Algorithms developed in his group for onboard crater recognition have been selected as the backup approach to regional hazard avoidance for the MSL mission. He also conducts research on terrain classification with a wide variety of sensors for off-road navigation on Earth. He was awarded the NASA Exceptional Achievement Medal in 2001 for his contributions to computer vision for space missions. He is an Adjunct Assistant Professor of Computer Science at the University of Southern California.